

Assignment #7: Nov Mock Exam立冬

Updated 1646 GMT+8 Nov 7, 2024

2024 fall, Compiled by 昂奕, 化学与分子工程学院

说明:

- 1) 月考: AC6。考试题目都在“题库（包括计概、数算题目）”里面，按照数字题号能找到，可以重新提交。作业中提交自己最满意版本的代码和截图。
- 2) 请把每个题目解题思路（可选），源码Python, 或者C++（已经在Codeforces/Openjudge上AC），截图（包含Accepted），填写到下面作业模版中（推荐使用 typora <https://typoraio.cn>，或者用word）。AC 或者没有AC，都请标上每个题目大致花费时间。
- 3) 提交时候先提交pdf文件，再把md或者doc文件上传到右侧“作业评论”。Canvas需要有同学清晰头像、提交文件有pdf、“作业评论”区有上传的md或者doc附件。
- 4) 如果不能在截止前提交作业，请写明原因。

1. 题目

E07618: 病人排队

sortings, <http://cs101.openjudge.cn/practice/07618/>

思路:

略

时间: 4min

代码:

```
n=int(input())
ren=[]
lao=[]
for i in range(n):
    id,age=input().split()
    age=int(age)
    if age>=60:
        lao.append([id,age])
    else:
        ren.append([id,age])
lao.sort(key=lambda x:x[1],reverse=True)
for i in lao:
    print(i[0])
for j in ren:
    print(j[0])
```

代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
n=int(input())
ren=[]
lao=[]
for i in range(n):
    id,age=input().split()
    age=int(age)
    if age>=60:
        lao.append([id,age])
    else:
        ren.append([id,age])
lao.sort(key=lambda x:x[1],reverse=True)
for i in lao:
    print(i[0])
for j in ren:
    print(j[0])
```

E23555: 节省存储的矩阵乘法

implementation, matrices, <http://cs101.openjudge.cn/practice/23555/>

思路:

(没多想...看着是个E题就直接笨办法做了) 直接转换回正常矩阵乘

时间: 10min

代码:

```
n,m1,m2=map(int,input().split())
mat1=[[0]*n for i in range(n)]
mat2=[[0]*n for i in range(n)]
for i in range(m1):
    r,c,e=map(int,input().split())
    mat1[r][c]=e
for i in range(m2):
    r,c,e=map(int,input().split())
    mat2[r][c]=e
ansmat=[[0]*n for i in range(n)]
ans=[]
for i in range(n):
    for j in range(n):
        ansmat[i][j]=sum(mat1[i][k]*mat2[k][j] for k in range(n))
        if ansmat[i][j]!=0:
            ans.append([i,j,ansmat[i][j]])
for j in ans:
    print(*j,sep=' ')
```

代码运行截图 == (至少包含有"Accepted") ==

状态: Accepted

源代码

```
n,m1,m2=map(int,input().split())
mat1=[[0]*n for i in range(n)]
mat2=[[0]*n for i in range(n)]
for i in range(m1):
    r,c,e=map(int,input().split())
    mat1[r][c]=e
for i in range(m2):
    r,c,e=map(int,input().split())
    mat2[r][c]=e
ansmat=[[0]*n for i in range(n)]
ans=[]
for i in range(n):
    for j in range(n):
        ansmat[i][j]=sum(mat1[i][k]*mat2[k][j] for k in range(n))
        if ansmat[i][j]!=0:
            ans.append([i,j, ansmat[i][j]])
for j in ans:
    print(*j,sep=' ')
```

M18182: 打怪兽

implementation/sortings/data structures, <http://cs101.openjudge.cn/practice/18182/>

思路:

把技能按使用时间和攻击力大小排序，一个个用

时间: 12min

代码:

```
t=int(input())
for _ in range(t):
    n,m,b=map(int,input().split())
    jn=[]
    for i in range(n):
        t,x=map(int,input().split())
        jn.append((t,x))
    jn.sort(key=lambda x:(x[0],-x[1]))
    r=0
    for i in range(len(jn)):
        if i==0 or jn[i][0]==jn[i-1][0]:
            r+=1
        else: r=1
        if r <= m:
            b -= jn[i][1]
            if b<=0:
                print(jn[i][0])
                break
        else:continue
    else:print('alive')
```

代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
t=int(input())
for _ in range(t):
    n,m,b=map(int,input().split())
    jn=[]
    for i in range(n):
        t,x=map(int,input().split())
        jn.append((t,x))
    jn.sort(key=lambda x:(x[0],-x[1]))
    r=0
    for i in range(len(jn)):
        if i ==0 or jn[i][0]==jn[i-1][0]:
            r+=1
        else: r=1
        if r <= m:
            b -= jn[i][1]
            if b<=0:
                print(jn[i][0])
                break
        else:continue
    else:print('alive')
```

M28780: 零钱兑换3

dp, <http://cs101.openjudge.cn/practice/28780/>

思路:

上课讲的例题...

时间: 11min

代码:

```
import bisect
n,m=map(int,input().split())
v=list(map(int,input().split()))
v.sort()
dp=[0]*(m+2)
for i in range(1,m+1):
    tmpv=v[0:bisect.bisect(v,i)]
    try:
        dp[i]=min(dp[i-j] for j in tmpv)
    except ValueError:
        dp[i]=1e9
    dp[i]+=1
if dp[m]>=1e9:print(-1)
else:print(dp[m])
```

代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
import bisect
n,m=map(int,input().split())
v=list(map(int,input().split()))
v.sort()
dp=[0]*(m+2)
for i in range(1,m+1):
    tmpv=v[0:bisect.bisect(v,i)]
    try:
        dp[i]=min(dp[i-j] for j in tmpv)
    except ValueError:
        dp[i]=1e9
    dp[i]+=1
if dp[m]>=1e9:print(-1)
else:print(dp[m])
```

T12757: 阿尔法星人翻译官

implementation, <http://cs101.openjudge.cn/practice/12757>

思路:

把输入从million和thousand处分别切开，分成三个部分，对每个部分分别转换，然后位值原理相加

时间: 23min（还WA了一次，因为乘100万少敲了一个0）

代码:

```
zhb={'zero':0,'one':1,'two':2,'three':3,'four':4,'five':5,'six':6,'seven':7,
     'eight':8,'nine':9,'ten':10,
     'eleven':11,'twelve':12,'thirteen':13,'fourteen':14,'fifteen':15,'sixteen':16,'seventeen':17,
     'eighteen':18,'nineteen':19,'twenty':20,'thirty':30,'forty':40,'fifty':50,'sixty':60,
     'seventy':70,
     'eighty':80,'ninety':90,'hundred':100,'thousand':1000,'million':1000000}
def conv(l):
    if 'hundred' in l:
        h=l.index('hundred')
        l1=l[:h]
        l2=l[h+1:]
        ans=0
        ans+=sum(zhb[i] for i in l1)*100+sum(zhb[i] for i in l2)
    else:ans=sum(zhb[i] for i in l)
    return ans
line=input().split()
if 'negative' in line:
    line=line[1:]
    neg=-1
else:neg=1
if 'million' in line:
    m=line[:line.index('million')]
    line=line[line.index('million')+1:]
else:m=[]
if 'thousand' in line:
```

```
k=line[:line.index('thousand')]
line=line[line.index('thousand')+1:]
```

代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
zhb={'zero':0, 'one':1, 'two':2, 'three':3, 'four':4, 'five':5, 'six':6,
    'eight':8, 'nine':9, 'ten':10,
    'eleven':11, 'twelve':12, 'thirteen':13, 'fourteen':14, 'fifteen':15, 'sixteen':16,
    'eighteen':18, 'nineteen':19, 'twenty':20, 'thirty':30, 'forty':40, 'fifty':50,
    'seventy':70,
    'eighty':80, 'ninety':90, 'hundred':100, 'thousand':1000, 'million':1000000}

def conv(l):
    if 'hundred' in l:
        h=l.index('hundred')
        l1=l[:h]
        l2=l[h+1:]
        ans=0
        ans+=sum(zhb[i] for i in l1)*100+sum(zhb[i] for i in l2)
    else:ans=sum(zhb[i] for i in l)
    return ans

line=input().split()
if 'negative' in line:
    line=line[1::]
    neg=-1
else:neg=1
if 'million' in line:
    m=line[:line.index('million')]
    line=line[line.index('million')+1:]
else:m=[]
if 'thousand' in line:
    k=line[:line.index('thousand')]
    line=line[line.index('thousand')+1:]
else:k=[]
finalans=neg*(conv(m)*1000000+conv(k)*1000+conv(line))
print(finalans)
```

T16528: 充实的寒假生活

greedy/dp, cs10117 Final Exam, <http://cs101.openjudge.cn/practice/16528/>

思路:

最大不重叠区间问题，把区间按终止时间排序然后遍历

时间: 10min

代码:

```
n=int(input())
act=[]
for i in range(n):
    st,et=map(int,input().split())
    act.append((st,et))
act.sort(key=lambda X:X[1])
ans=1
end_time=act[0][1]
for i in range(len(act)):
    if act[i][0]<=end_time:
        continue
    else:
        end_time=act[i][1]
        ans+=1
print(ans)
```

代码运行截图（至少包含有"Accepted"）

状态: Accepted

源代码

```
n=int(input())
act=[]
for i in range(n):
    st,et=map(int,input().split())
    act.append((st,et))
act.sort(key=lambda X:X[1])
ans=1
end_time=act[0][1]
for i in range(len(act)):
    if act[i][0]<=end_time:
        continue
    else:
        end_time=act[i][1]
        ans+=1
print(ans)
```

2. 学习总结和收获

如果作业题目简单，有否额外练习题目，比如：OJ“计概2024fall每日选做”、CF、LeetCode、洛谷等网站题目。

周末准备把选做补上...一直在复习高数
期末要是也是这个难度我也许就可以把高数可以预见的爆炸的绩点拉回来一点了...呜呜呜